

Сервис расписания доставки сим-карт Подключи

<b>Сервис расписания доставки сим-карт Подключи</b>	<b>1</b>
<b>1. Руководство по развертыванию сервиса</b>	<b>2</b>
1.1. Создание окружения	2
1.2. Установка зависимостей	2
1.4. Создание таблиц бд	3
1.5. Запуск	3
1.5.1. Запуск сервера	3
1.5.2. Запуск модульных тестов загрузки файлом (тесты разметки):	3
<b>2. Руководство по использованию API</b>	<b>4</b>
2.1. Доступ	4
2.2. Методы API	4
2.2.3. Загрузить слоты доставки файлом	4
2.2.4. Получить список временных слотов для доставки	5
2.2.5. Получить слоты доставки файлом	5
2.2.6. Пример ответов с ошибкой (во всех методах):	5

# 1. Руководство по развертыванию сервиса

## 1.1. Создание окружения

```
sudo -E python3 -m venv ./
```

## 1.2. Установка зависимостей

```
sudo -E yum install gmp-devel
```

```
sudo rm -rf /tmp/pip-*
```

```
export LD_LIBRARY_PATH="/usr/lib64:$LD_LIBRARY_PATH"
```

```
sudo -E yum install python-devel
```

```
sudo -E bin/python -m pip install -r requirements.txt
```

## 1.3. Настройки

создать файл settings.py в корне проекта

пример файла настроек с необходимыми параметрами:

```
# settings.py
DEBUG = True
PROCESSES_NUM = 1
DB_PATH = 'postgresql://<db-
user>:<password>@<db_ip>:<db_port>/<db_name>'

SERVER_SCHEMA = 'https'
SERVER_HOST = '<server url>'

PORT = SERVER_PORT = 9008

SERVER_URL = '{}://{}'.format(SERVER_SCHEMA, SERVER_HOST)

SERVER_AUTH_TOKENS = {
    '<токен для чтения>': 'read',
    '<токен для записи>': 'write',
}
```

## 1.4. Создание таблиц бд

bin/python sync\_db.py

## 1.5. Запуск

### 1.5.1. Запуск сервера

bin/python server.py

### 1.5.2. Запуск модульных тестов загрузки файлом (тесты разметки):

bin/python tests.py

## 2. Руководство по использованию API

### 2.1. Доступ

Методы API вызываются с заголовком `ServerAuthorization`, в котором должен быть токен доступа на запись либо на чтение

Пример вызова:

```
curl -X POST -H "ServerAuthorization: "<access_token>" -H "Content-Length: 116" -d '{"region_id": "450000000000", "local_time": "2021.04.01T16:10:00", "subject": "Москва"}' 'http://10.77.35.96:9028/api/v1/delivery_date_times'
```

### 2.2. Методы API

#### 2.2.3. Загрузить слоты доставки файлом

`POST /api/v1/upload_dates.xls`

`->`

`<xls или xlsx файл>`

`<-`

```
{
  updated_delivery_dates_count: int, // обновлено записей
  added_delivery_dates_count: int // добавлено записей
}
```

## 2.2.4. Получить список временных слотов для доставки

POST /api/v1/delivery\_date\_time

```
->
{
  region_id: string // код окато (необязательный параметр) #
'410000000000'
  local_time: string // день для которого необходимо расписание #
'2021.02.14T16:10:00'
  subject: string // наименование города (необязательный параметр) #
'Москва'
}
```

Без уточнения региона и города (subject, region\_id):

```
<-
{
  "result": 1,
  "error": 0,
  "delivery_info_text": {okato: text, ...}, // текст для отображения
  "delivery_prices": {okato: price, ''}, // цены на доставку
  "default_region": string, // Код ОКАТО по-умолчанию
  "delivery_regions": [{ 'id' : string, 'name': string}] // список
регионов с наименованиями городов
}
```

С регионом и городом (subject, region\_id):

```
<-
{
  "result": 1,
  "error": 0,
  "delivery_times": {"DD.MM.YYYY": ["HH:MM", "HH:MM", ...], ...}, //
слоты доставки
  "delivery_regions": [{ 'id' : string, 'name': string}] // список
регионов с наименованиями городов
  'price': int // цена дотавки в указанном регионе
}
```

## 2.2.5. Получить слоты доставки файлом

POST /api/v1/dates.xls

```
<- <xlsx файл>
```

## 2.2.6. Пример ответов с ошибкой (во всех методах):

```
{
  'error': 1,
  'reply': 'Текст ошибки'
}
```